Experiments on speaker tracking and segmentation in radio broadcast news

Daniel Moraru, Mathieu Ben, Guillaume Gravier

Institut de Recherche en Informatique et Systèmes Aléatoires (CNRS & INRIA)

Equipe Modélisation et Expérimentation pour le Traitement de l'Information et du Signal Sonores

http://www.irisa.fr/metiss

{daniel.moraru,mathieu.ben,guillaume.gravier}@irisa.fr

Abstract

In this paper we describe the speaker tracking and clustering system that we implemented for the ESTER evaluation campaign. We present some experiments on normalization in speaker tracking, in particular concerning the use of t-norm for speaker tracking in broadcast news. Results show that the use of t-norm significantly improves the performance at low false alarm rates. In a second part of the paper, we study the possible interactions between speaker tracking and speaker segmentation (also known as speaker diarization). We show that speaker segmentation benefits from the use of speaker tracking as a prior information while the contrary is not true. Using speaker tracking before clustering can decrease the speaker segmentation error by 4% absolute.

1. Introduction

Speaker segmentation and speaker tracking are two important tasks of spoken document indexing. The speaker segmentation tasks consists in identifying all regions of a document that were uttered by the same speaker. Identification of speakers is not required and no prior knowledge about the number of speakers in the document or about the speakers characteristics is provided. On the contrary, speaker tracking aims at detecting regions uttered by a given speaker for which training data is available beforehand.

In most approaches, both the speaker segmentation and the speaker tracking tasks can be divided into three main steps where the two first steps are common to both tasks. The first step consists in detecting portions of the document containing speech while the second step aims at detecting speaker and acoustic changes in the speech portions. The segmentation thus obtained is then used for speaker tracking and segmentation.

In the case of speaker tracking, a speaker verification algorithm, such as [1], can be used to detect whether the target speaker is present or not in each of the segments.

In speaker segmentation, the aim of the third step is to determine the actual number of speakers and to group together segments from the same speaker. A commonly used approach is hierarchical clustering, often implemented in a bottom-up manner, with a stop criterion to determine the number of speakers. In this approach, a distance between clusters is used to find out the two closest clusters which are merged if this results in a better segmentation according to some quality measure. The Bayesian information criterion is used as a stop criterion as well as a distance criterion between clusters in many studies. Typical systems implementing such features can be found in [2, 3]. Some variants of this classical approach include a global optimization criterion which optimize speaker change detection jointly with speaker clustering [4], or the use of a resegmentation step at some point in the clustering to improve the speaker change detection [5].

The two tasks are usually carried out independently. However, in [6], speaker models are used in a clustering algorithm to determine the purity of a cluster. In this study, speaker identification is carried out on clustered segments. Clusters with a high speaker purity are tagged with the best model while clusters with a low speaker purity are split according to the speaker identification result.

In this paper, we first give an overview of the baseline speaker segmentation and tracking system that we implemented for the French broadcast news rich transcription campaign ESTER [7]. In particular, we study the influence of t-norm [8] on data where the channel variability is not a major factor and validate our GMM-based approach of speaker segmentation previously described in [9]. In a second part of the paper, we investigate how the speaker segmentation and tracking tasks can be combined to improve performances. We first investigate how segmentation can help tracking before investigating the impact of speaker tracking on the segmentation.

2. Experimental setup

Experiments were carried in the framework of the French evaluation campaign of broadcast news rich transcription systems ESTER [7]. We briefly recall here the content of the corpus as well as the evaluation rules for the speaker segmentation and tracking tasks.

2.1. Corpus

The entire corpus consists of a training set of 90 hours, a development set of 8 hours and a test set of 10 hours. The training and development sets include four radio stations and were recorded in contiguous periods of time. The test set contains two additional radio stations, recorded more than a year after the development set.

Apart from the time period gap between the development set and the test set, the latter is more difficult as it contains more interviews and more speech in degraded conditions. There are also many speaker differences between the two sets. For those reasons, we divided the development set into two sets of four hours each, respectively dev1 and dev2, the former preceding the latter in time by a couple of days. The dev1 set is used as the development set while the dev2 set is used as an additional evaluation set.

Each of the dev1 and dev2 sets contain 7 shows with approximately 600 turns and 140 speakers. The test set contains 18 shows, 341 speakers with a total of 2,137 turns. The average

length of a turn is around 20s on all the sets.

2.2. Evaluation rules

Evaluation rules for the speaker segmentation task specify that segmentation is relative to the shows. Therefore grouping of segments from the same speaker but in different shows is not required. Performance is evaluated after an optimal mapping of the arbitrary speaker names from the segmentation system and the true speaker names. The error rate is proportional to the amount of speech that was mistakenly attributed to a speaker, plus the amount of missed speech and the amount of inserted speech.

For the tracking task, a list of 279 target speakers with at least 2 minutes of speech in the training set was provided. The amount of training speech available for a target speaker varies from 2 minutes to about one hour. For some speakers, typically anchor persons, information on the associated broadcaster(s) were also provided. The task consisted in tracking every listed speaker in every show. Performance is related to the amount of speech wrongly detected as uttered by the target speaker (false alarm) and to the amount of speech from the target speaker that was not detected (miss). For the evaluation campaign, performance was measured using the F-measure. In this paper, we compare systems using detection error trade-off (DET) curves which plot the miss rate against the false alarm rate.

In the dev1 and dev2 sets, respectively 63% and 55% of the total amount of speech is due to around 40 speakers of the target speaker list. In the test set, the ratio drops to 20% of the total amount of speech due to 33 speakers of the target speaker list.

3. Baseline system

In this section, we described the baseline system developed for the evaluation campaign and report on some experiments made on the speaker tracking system.

3.1. Speech and speaker change detection

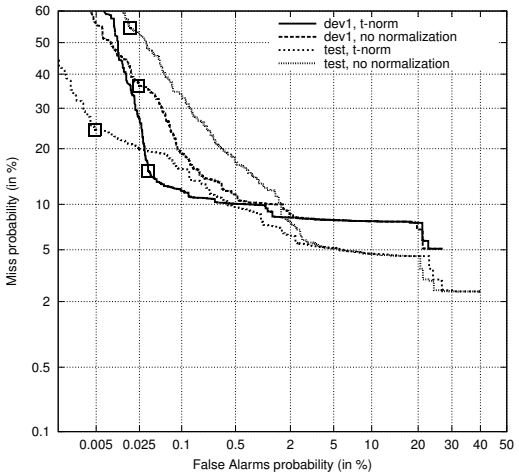
Speech detection is carried out with a 4 state ergodic hidden Markov model where the states represent speech, speech with background music, music and silence. State conditional probabilities are modeled by a 256 component mixture of Gaussians with diagonal covariance. The feature vector is a 25 component vector with 12 cepstral coefficients plus the first and second order derivatives with a mean and variance normalization. After Viterbi decoding, all segments containing speech, whether with background music or not, are tagged as speech and adjacent speech segments are merged. The miss and inserted speech rates are respectively 1.25% and 5.7% on the development set, and 1.9% and 14% on the test set.

Speaker change detection is carried out independently in each of the detected speech segments with a three pass variant of the BIC change detection algorithm [3] with 24-channel Mel filter-bank features and a full covariance matrix. The average segment length at the output of the speaker change detection algorithm is around 10 seconds, with a purity of about 98%.

3.2. Tracking

The tracking system is based on a classical GMM-UBM approach [1] where speaker detection is performed individually on each speech segment, with feature vectors containing 16 cepstral coefficients, their Δ coefficients and the Δlog energy. For sake of rapidity, speakers for which a broadcaster is identified are tracked only in the relevant shows. Experiments showed no performance improvement using the broadcaster knowledge compared to tracking all speakers in all the shows.

Figure 1: Detection error trade-off curves on the dev1 and test sets with and without t-norm. Squares indicate the maximum F-measure operating point.



A universal background model (UBM) is estimated using 13 hours of speech from the training set. This model is a 512 component Gaussian mixture model (GMM) resulting from the combination of two gender-dependent GMM with 256 components each. Target speaker models are adapted from the UBM using a maximum a posteriori (MAP) criterion, where only the mean vectors are adapted with a single EM iteration.

The raw detection score for a segment is the average log-likelihood ratio (LLR) per frame, calculated on the whole segment and computed using the 10 best gaussians of the UBM for each frame. The raw scores are then t-normalized [8] using a cohort of impostor models (117 female and 133 male speakers) taken from the training corpus and distinct from either the target speakers or the UBM speakers. Decision thresholds were optimized on the dev1 set so as to maximize the F-measure.

Performance with and without t-norm are plotted in Figure 1 for the dev1 and test sets. At the actual operating point, this system achieves a false alarm rate and miss rate of respectively 0.08% and 13.8% on the dev1 set, and 0.005% and 24% on the test set.

An interesting result is that t-norm significantly improves the performance at low false alarm rates, in particular on the test set. This is surprising since t-norm is supposed to compensate mainly for mismatch by making the LLR score relative to the average test segment score on impostor models. However, in broadcast news material, there is strictly speaking no acoustic condition mismatch. Moreover, since segments provided by the speaker change detection algorithm are quite long, the phonetic content of the segment should be rich enough to not require t-norm compensation. Also, t-norm is much more efficient on the test set than on the dev1 set, probably because of slightly different recording conditions between the two sets.

Table 1: Speaker segmentation error rate on the dev1, dev2 and test sets

Corpus	dev1	dev2	test
error	18.9%	14.9%	17.9%

3.3. Clustering for speaker segmentation

The speaker segmentation system implements a bottom-up clustering with a global BIC stop criterion, where clusters are modeled using 32 component GMM. Feature vectors consist of 16 mel frequency cepstral coefficients plus energy.

For a given show, clusters are first initialized from the segments by adapting the mean vectors of a generic document speech background model (DSBM) according to a MAP criterion. The DSBM parameters are estimated using the speech data from the entire show. A model-space based approximation of the Kullback-Liebler divergence between two GMM whose mean vectors were adapted from the same model [9], is used as a distance measure between two clusters. The main advantage of this distance based solely on the parameters of two GMM is its low computation time compared to a generalized likelihood ratio. The two closest clusters are merged until a stop criterion is reached. The clustering stop criterion is based on the detection of a global maximum of the global BIC criterion, with

$$\mathrm{BIC}(M) = \log L(y|M) - \lambda \frac{m}{2} N_{\mathrm{sp}} \log N_y \qquad (1)$$

where $L(y|M)$ is the likelihood of the entire data given the current cluster models. The global BIC criterion (1) has the advantage that the number of speakers N_{sp} appears explicitly in the penalization term. The coefficient λ is tuned on the dev1 set and applied as is on the dev2 and test sets.

Results are presented in Table 1. In addition, the best result that could be obtained on the dev1 set, with coefficients λ optimized separately on each file, is 16.8%, compared to 18.9% with a unique λ.

4. Using clustering for tracking

In a first experiment to combine the speaker segmentation and tracking tasks, we investigate how clustering can improve tracking.

By construction, bottom-up clustering algorithms increase the speaker purity and decrease the cluster purity across iterations. Indeed, as the clustering proceeds, the data from one speaker tend to be in the same cluster while clusters tend to receive data from various speakers. However, the typical behavior of such algorithms is that the cluster purity remains high for a long time while, in the meantime, speaker purity increases.

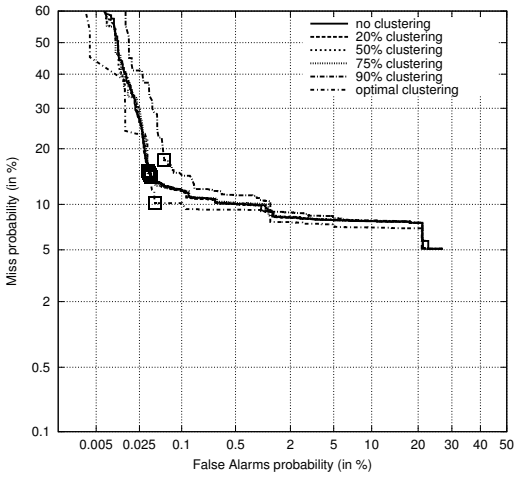
Therefore, the idea is to run a few iterations of clustering before performing speaker tracking on the clusters rather than individually on each segment. The motivation for this is to try to increase the amount of data available in the clusters without deteriorating their purity, in the hope that the tracking system makes better decisions with more data available.

Table 2 shows the average cluster purity and length as a function of the reduction of the original number of segments (*i.e.* a reduction of 50% means that the number of clusters is half the number of segments before clustering) for the dev1 set. Speaker tracking results are reported for the dev1 set in Figure 2 for different amount of reduction as well as for an optimal segmentation. The optimal segmentation corresponds to the best clustering of the automatically derived segments. It is obtained by comparing the segments given by the automatic speaker change detection algorithm to the true speaker segmentation.

Table 2: Average cluster purity (in %) and length (in sec.) as a function of the amount of clustering.

	none	20%	50%	75%	90%	optimal
purity	98.0	97.2	96.4	94.8	86.4	98.0
length	9.7	12.1	19.4	39.1	99.3	87.4

Figure 2: Detection error trade-off curves on the dev1 set for various amount of automatic clustering and for optimal clustering. Squares indicate the maximum F-measure operating point.



Results clearly show that, though the average cluster length increases without any significant drop of the average purity, tracking does not benefit from clustering. Tracking results with an optimal segmentation show that a marginal gain could be obtained, should the clustering algorithm be perfect. However, the best possible gain is small. This may be partly due to the fact that the tracking system already performs well on initial segments with an average length of 10 seconds. It may also be partly due to the fact that the clustering algorithm groups together segments on which the tracking system makes the same decision, which is thus not affected by the grouping.

An interesting point to note is that tracking performance significantly degrades only after a large amount of clustering (more than 85% reduction) is performed. This could be interesting to exploit in conjonction with a system using idiolectal features.

5. Using tracking for clustering

In this section we investigate the use of the result of the speaker tracking system described in 3.2 as *prior* information for clustering.

The idea is to use the output of the tracking algorithm as a starting point for clustering. Every segment that is not identified by the speaker tracking system is treated as a single class. Every group of segments identified by the speaker tracking system is considered as a single class. During clustering merging of identified classes is not allowed.

Figure 3: Speaker segmentation error rate, speaker tracking false alarm rate (x 100) and miss rate on the dev1 set as a function of the speaker tracking decision threshold.

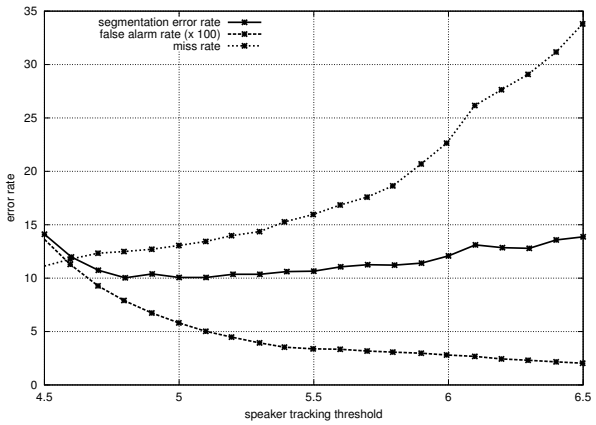


Table 3: Speaker segmentation error on the dev1, dev2 and test sets with local and global coefficients λ

	with tracking		no tracking
λ	local	global	global
dev1	10.1%	10.9%	19.0%
dev2	8.8%	10.1%	14.9%
test	13.6%	19.2%	17.9%

The optimal operating point for the tracking system was tuned on the dev1 set so as to minimize the optimal segmentation error rate. For every possible operating point, the best possible segmentation error was computed, as in section 3.3 with coefficients λ optimized independently on each file. Figure 3 shows this optimal segmentation error rate as a function of the decision threshold in the tracking system.

Results obtained with a global λ tuned on the dev1 test are given in Table 3 and compared to the results obtained with file dependent λ's. These results show that the use of the speaker tracking result not only significantly decreases the segmentation error rate but also decreases the difference between the local and the global λ. This means we are getting closer to the optimal clustering result. This is somehow normal since the identified classes issued from speaker tracking are bigger and usually pure, thus resulting in a better estimation of the likelihood term in (1).

Results obtained on the dev1 set are confirmed on the dev2 set. However as we can see the use of speaker tracking slightly degrades the performances on the test set. The explanation resides in the fact that the amount of speech belonging to the tracking target speakers is significantly smaller in the test set than in the dev1 and dev2 sets, as noted in section 2.2. In this case, speaker tracking should at most slightly improve the segmentation performance.

6. Conclusion

In this paper we have investigated the use of the t-norm for speaker tracking in boradcast news and the interaction between the speaker segmentation and speaker tracking modules of our indexing system. Our experiments were conducted on the French ESTER radio broadcast news corpus.

Experiments on score normalization showed that t-norm significantly improves performance at low false alarm rates even though there is less acoustic mismatch in broadcast news shows compared to telephone speech for exemple.

A second set of experiments explored the interaction between speaker tracking and speaker segmentation. Results demonstrated that clustering does not improve speaker tracking, even with a wizzard clustering result. The reason is most likely twofold: (*i*) the speaker tracking system already performs well on the initial segments and (*ii*) the clustering groups together segments on which the tracking system has the same behavior. On the contrary, an important improvement was obtained when using speaker tracking to provide prior information for the speaker segmentation algorithm. However, this improvement depends on the amount of speech due to known speakers. When this amount is too small, performance slightly decrease.

A straightforward extension of this work is to design an interactive tracking and clustering approach in order to improve both speaker segmentation and tracking result.

7. References

[1] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1–3, 2000.

[2] M. Siegler, U. Jain, B. Raj, and R. Stern, "Automatic segmentation, classification and clustering of broadcast news audio," in *DARPA Speech Recognition Workshop*, 1997.

[3] S. Chen and P. S. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the Bayesian information criterion," in *Broadcast News Transcription and Understanding Workshop*, 1998, pp. 127–132.

[4] J. Ajmera and C. Wooters, "A robust speaker clustering algorithm," in *IEEE Workshop on Automatic Speech Recognition Understanding*, 2003.

[5] C. Barras, X. Zhu, S. Meigner, and J.-L. Gauvain, "Improving speaker diarization," in *Proc. DARPA Rich Transcription 04*, November 2004.

[6] P. Deléglise, Y. Estève, B. Jacob, T. Merlin, and S. Meigner, "Campagne d'évaluation ESTER 2005 : Transcription et segmentation en locuteurs." in *Proc. of the ESTER Phase II workshop*, 2005.

[7] G. Gravier, J. F. Bonastre, S. Galliano, E. Geoffrois, K. M. Tait, and K. Choukri, "The ESTER evaluation campaign of rich transcription of french broadcast news," in *Language Evaluation and Resources Conference*, 2004.

[8] R. Auckenthaler, M. Carey, and H. Lloyd-Thomas, "Score normalization for test-independent speaker verification systems," *Digital Signal Processing*, vol. 10, no. 1–3, 2000.

[9] M. Ben, M. Betser, F. Bimbot, and G. Gravier, "Speaker diarization using bottom-up clustering based on a parameter-derived distance between adapted gmms," in *Intl. Conf. on Speech and Language Processing*, 2004.